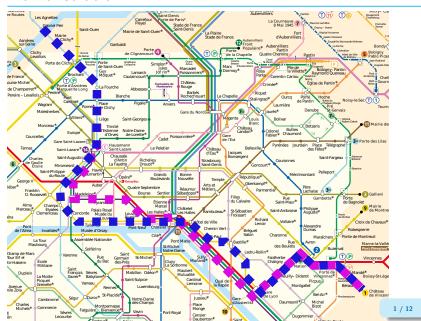
PROJET C++, ENSAE



Itinéraire entre deux stations du métro parisien

KIM ANTUNEZ ET ALAIN QUARTIER-LA-TENTE 07/01/2020 - 15h30 à 15h45 Ensae — 2019-2020

Démonstration

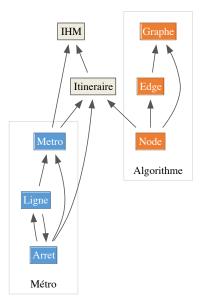


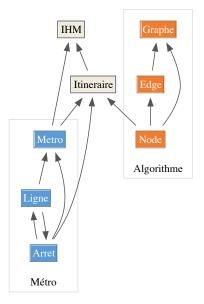
Sommaire

1. Description des classes

2. L'algorithme de Dijkstra

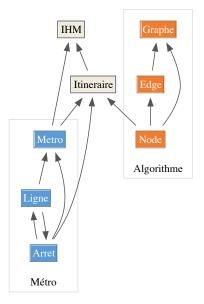
3. Pistes d'amélioration



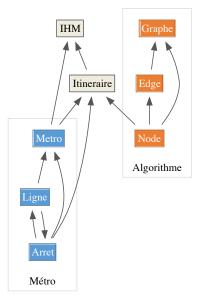


8 classes regroupées en 4 groupes modulables :

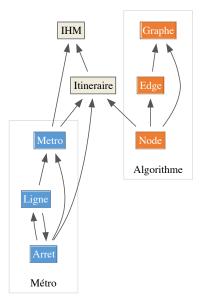
 Les données : Arret, Ligne et Metro



- Les données : Arret, Ligne et Metro
- L'algorithme : Nodge, Edge et Graphe

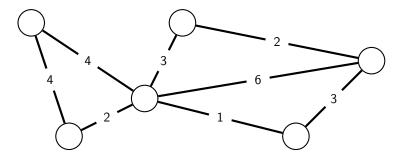


- Les données : Arret, Ligne et Metro
- L'algorithme : Nodge, Edge et Graphe
- Lien entre les données et l'algorithme : **Itineraire**

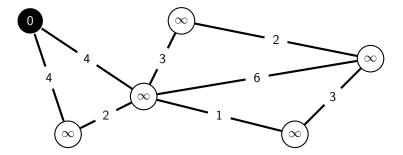


- Les données : Arret, Ligne et Metro
- L'algorithme : Nodge, Edge et Graphe
- Lien entre les données et l'algorithme : **Itineraire**
- L'interface : IHM

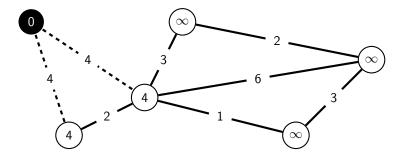
L'algorithme de Dijkstra (1/8)



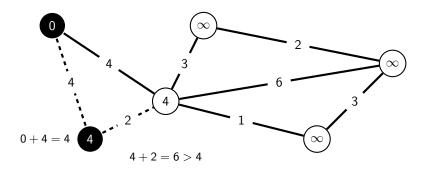
L'algorithme de Dijkstra (2/8)



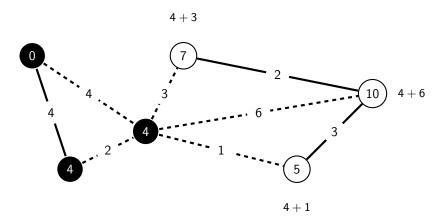
L'algorithme de Dijkstra (3/8)



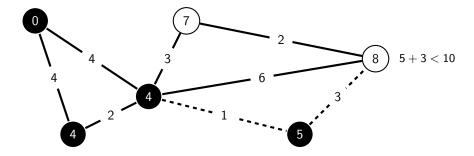
L'algorithme de Dijkstra (4/8)



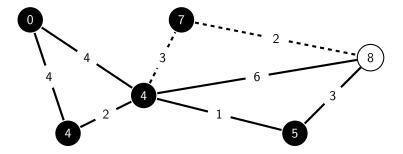
L'algorithme de Dijkstra (5/8)



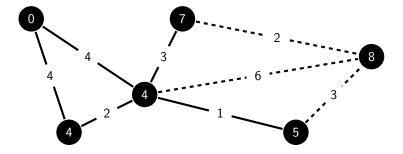
L'algorithme de Dijkstra (6/8)



L'algorithme de Dijkstra (7/8)



L'algorithme de Dijkstra (8/8)





Pistes d'amélioration

- 1. Améliorer le chargement des données
- 2. Prendre en compte l'horaire des métros (ajout des temps de passage dans Edge)
- Prendre en compte le réseau de bus, de RER et de tramways, le temps de marche entre deux stations...
- 4. Proposer plusieurs itinéraires quand ceux-ci ont un temps de trajet proche
- 5. Ajouter d'autres types d'itinéraires : accessibles aux personnes en situation de handicap, passant par des toilettes publiques. . .